

UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



TRABAJO FIN DE MÁSTER

Desarrollo aplicación web de autorregistros para terapeutas psicológicos y sus pacientes

Máster Universitario en Ingeniería de Telecomunicación

Autor: Rico Maldonado, Juan
Tutor: Boemo Scalvinoni, Eduardo

FECHA: junio, 2020

Desarrollo aplicación web de autorregistros para terapeutas psicológicos y sus pacientes

AUTOR: Rico Maldonado, Juan
TUTOR: Boemo Scalvinoni, Eduardo

Dpto. Tecnología Electrónica y de Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Junio de 2020

Resumen

En este Trabajo de Fin de Máster se va a implementar una aplicación web que proporcione una nueva herramienta para psicólogos clínicos y psiquiatras. Ésta permitirá diseñar y ofrecer evaluaciones psicológicas online para pacientes de modo remoto. Así, es posible que éstos reciban una atención clínica personalizada entre sesiones psicoterapéuticas. También facilitará el seguimiento tras finalizar la terapia.

La aplicación desarrollada permitirá que los pacientes tengan a su alcance una herramienta para realizar autoevaluaciones y ejercicios conductuales diseñados por sus terapeutas para ser llevados a cabo en los períodos entre sesiones, fuera de terapia, así como una herramienta de visualización de resultados por el terapeuta para que puedan ser analizados y explicados al paciente durante la consulta.

La herramienta que se propone en este TFM tendrá múltiples usos: se podrá usar para monitorizar sintomatología afectiva ante situaciones relevantes, interacciones sociales, ocurrencia de pautas de consumo problemático (ej. número de cigarrillos consumidos, frecuencia diaria en consumo de alcohol), con el fin de realizar un análisis funcional y diseñar planes específicos de tratamiento.

Abstract

In this Master's thesis, a web application will be implemented to provide a new tool for clinical psychologists and psychiatrists. It will allow designing and offering online psychological evaluations for remote patients. Thus, it is possible for them to receive personalized clinical attention between psychotherapy sessions. It will also facilitate follow-up after therapy is completed.

The application developed will allow patients to have at their disposal a tool to perform behavioral assessments and exercises designed by their therapists to be carried out in the periods between sessions, outside therapy, as well as a tool for visualization of results by the therapist so that they can be analyzed and explained to the patient during the consultation.

The tool proposed in this TFM will have multiple uses: it can be used to monitor affective symptoms in relevant situations, social interactions, the occurrence of problematic consumption patterns (e.g. number of cigarettes consumed, daily frequency of alcohol consumption), in order to perform a functional analysis and design specific treatment plans.

Palabras clave

Autorregistro, autoinforme, online, psicología, terapeuta, paciente, XAMPP, Bitnami, AWS.

Keywords

Self-reporting, online, psychology, therapist, patient, XAMPP, Bitnami, AWS.

Agradecimientos

A mi familia, mis amigos, y mis tutores Eduardo y Álvaro.

ÍNDICE DE CONTENIDOS

1 INTRODUCCIÓN.....	1
1.1 MOTIVACIÓN	1
1.2 OBJETIVOS.....	2
2 ESTADO DEL ARTE	5
2.1 INTRODUCCIÓN: AUTOINFORMES	5
2.2 AUTORREGISTROS	6
2.2.1 <i>Autorregistros en dispositivos móviles</i>	7
2.3 APLICACIÓN WEB	9
2.3.1 <i>HTML</i>	9
2.3.2 <i>Métodos GET y POST</i>	9
2.3.3 <i>PHP</i>	10
2.3.4 <i>MySQL</i>	10
2.3.5 <i>phpMyAdmin</i>	11
2.3.6 <i>CSS</i>	11
2.3.7 <i>JavaScript</i>	12
3 DISEÑO Y DESARROLLO	13
3.1 OBJETIVOS DE DISEÑO.....	13
3.1.1 <i>Diseño de la aplicación web</i>	13
3.2 PRIMEROS PASOS.....	15
3.2.1 <i>Instalación de XAMPP</i>	15
3.2.2 <i>Configuración de phpMyAdmin</i>	16
3.2.3 <i>Formularios HTML y acceso a la base de datos</i>	17
3.2.4 <i>Desarrollos formularios de registro e inicio de sesión</i>	18
3.3 DESARROLLO WEB DESTINADA A TERAPEUTAS	20
3.3.1 <i>Crear autorregistros</i>	20
3.3.2 <i>Lista de autorregistros</i>	22
3.3.3 <i>Lista de pacientes</i>	23
3.3.4 <i>Resultados</i>	24
3.3.5 <i>Perfil</i>	26
3.3.6 <i>Cerrar sesión</i>	26
3.4 DESARROLLO WEB DESTINADA A PACIENTES.....	26
3.5 ENCRIPCIÓN DE DATOS.....	28
3.5.1 <i>Encriptación con funciones hash</i>	28
4 INTEGRACIÓN Y RESULTADOS	31
4.1 INTEGRACIÓN.....	31
4.1.1 <i>Alojamiento en Amazon Cloud</i>	31
4.1.2 <i>Página de inicio</i>	34
4.1.3 <i>Cuestiones de interés legal</i>	35
4.1.4 <i>De HTTP a HTTPS</i>	35
5 CONCLUSIONES Y TRABAJO FUTURO.....	37
5.1 CONCLUSIONES.....	37
5.2 TRABAJO FUTURO	38
6 BIBLIOGRAFÍA	41
GLOSARIO	- 1 -

ÍNDICE DE ILUSTRACIONES

ILUSTRACIÓN 1: EJEMPLO AUTORREGISTRO SOBRE EL CONSUMO DE CIGARRILLOS.	7
ILUSTRACIÓN 2: EJEMPLO AUTORREGISTRO SITUACIONES QUE CAUSAN ESTRÉS.	7
ILUSTRACIÓN 3: ESTRUCTURA PÁGINA WEB DESTINADA A LOS TERAPEUTAS.	14
ILUSTRACIÓN 4: ESTRUCTURA PÁGINA WEB DESTINADA A LOS PACIENTES.	14
ILUSTRACIÓN 5 : PANEL DE CONTROL DE XAMPP.	15
ILUSTRACIÓN 6: INTERFAZ DE PHPMYADMIN.	16
ILUSTRACIÓN 7: EJEMPLO ASPECTO FORMULARIO HTML.	18
ILUSTRACIÓN 8: FORMULARIO REGISTRO PACIENTES.	19
ILUSTRACIÓN 9: FORMULARIO INICIO DE SESIÓN PACIENTES.	19
ILUSTRACIÓN 10: CREACIÓN DE AUTORREGISTROS.	20
ILUSTRACIÓN 11: CREACIÓN AUTORREGISTRO.	21
ILUSTRACIÓN 12: LISTA DE AUTORREGISTROS.	22
ILUSTRACIÓN 13: SELECCIÓN DE FECHAS.	23
ILUSTRACIÓN 14: LISTA DE PACIENTES.	23
ILUSTRACIÓN 15: EJEMPLO RESULTADOS.	24
ILUSTRACIÓN 16: EJEMPLO PÁGINA RESULTADOS.	25
ILUSTRACIÓN 17: MENÚ APLICACIÓN DESTINADA A PACIENTES.	27
ILUSTRACIÓN 18: MENÚ APLICACIÓN DESTINADA A PACIENTES.	28
ILUSTRACIÓN 19: EJEMPLO ESTRUCTURA DE CONTRASEÑA ENCRYPTADA.	29
ILUSTRACIÓN 20: RESULTADO AL ACCEDER A HTTP://[NOMBRE_DEL_SERVIDOR]/PHPINFO.PHP.	32
ILUSTRACIÓN 21: CAPTURA FIELEZILLA.	33
ILUSTRACIÓN 22: CREDENCIALES BITNAMI.	33
ILUSTRACIÓN 23: PÁGINA DE INICIO.	34
ILUSTRACIÓN 24: CERTIFICADO SSL.	36

1 Introducción

1.1 Motivación

Este TFM es una colaboración con el proyecto de Investigación E-Motion Lab (<https://www.ucm.es/e-motionlab/>) de la Facultad de Psicología de la Universidad Complutense de Madrid (UCM) que está dirigido por el profesor Álvaro Sánchez López (Grupo de Investigación Consolidado en Psicopatología y Salud Mental Positiva) (<https://www.ucm.es/grupos/grupo/175>). El proyecto es financiado por el Programa de Atracción de Talento Investigador de la Comunidad de Madrid y el Plan Nacional de I+D (Programa “Generación de Conocimiento”) de la Agencia Estatal de Investigación. El trabajo de investigación de este equipo se centra en la utilización de nuevas tecnologías para la evaluación de procesos psicopatológicos, como los sesgos cognitivos y los problemas de regulación emocional, implicados en el inicio y el mantenimiento de trastornos emocionales como la depresión y la ansiedad.

El tema del TFM es colaborar desarrollando una aplicación web que proporcione una nueva herramienta para psicólogos clínicos y psiquiatras. Ésta permitirá diseñar y ofrecer evaluaciones psicológicas online para pacientes de modo remoto. Así, es posible que éstos reciban una atención clínica personalizada entre sesiones psicoterapéuticas. También facilitará el seguimiento tras finalizar la terapia.

La aplicación desarrollada permitirá que los pacientes tengan a su alcance una herramienta para realizar evaluaciones diseñados por sus terapeutas para ser llevados a cabo en los períodos entre sesiones, fuera de terapia.

El trabajo en psicoterapia va más allá de la realización de entrevistas clínicas y aplicación de cuestionarios y técnicas terapéuticas en sesión. Por el contrario, parte esencial del abordaje psicológico se refiere al uso de procedimientos de autorregistro (Pérez Álvarez & Fernández-Ballesteros, 1994). Estas técnicas consisten en lo siguiente:

- Técnicas semiestructuradas en las que al sujeto-paciente se le suministra una hoja de papel en la que figuran específicamente conductas que deben anotarse y las condiciones en las que deben hacerse, y es el sujeto-paciente el que cumplimenta enteramente tal protocolo, fuera de sesión.
- Generalmente, las conductas se anotan en el mismo momento de producirse o, a veces, para evitar problemas de reactividad o bien por el tipo de objeto de evaluación, se pide al sujeto-paciente que las registre de forma diferida, es decir, tras ocurrir el evento.
- Se trata de técnicas utilizadas en situaciones naturales, durante la vida diaria del evaluado.
- Tal información registrada en papel se trae entonces a la siguiente sesión de terapia, de modo que terapeuta y paciente analizan dichos resultados, trabajan sobre ellos y

definen objetivos de autorregistro o intervención para el siguiente período entre sesiones.

Dichas técnicas tienen múltiples usos (Fernández-Ballesteros, 2011): se pueden usar para monitorizar sintomatología afectiva ante situaciones relevantes, interacciones sociales, ocurrencia de pautas de consumo problemático (por ejemplo, número de cigarrillos consumidos, frecuencia diaria en consumo de alcohol), problemáticas de alimentación, número de horas de sueño, o incluso requerir la observación de los eventos antecedentes y consecuentes de conducta objetivo con el fin de realizar un análisis funcional y diseñar planes específicos de tratamiento.

Este trabajo es, por tanto, esencial durante el tratamiento terapéutico psicológico, sin embargo, el uso de métodos de lápiz y papel plantea serias desventajas, incluyendo:

- Dificultades en el diseño y planificación de cada tipo de autorregistro específico por el terapeuta y el entrenamiento del paciente en su utilización fuera de consulta.
- Frecuentes problemas del paciente a la hora de recordar llevar encima todos los materiales en papel, no registrándose las conductas en el momento adecuado.
- Problemas del paciente para encontrar un lugar y momento adecuados para completar los autorregistros en papel.
- Frecuentes olvidos del paciente a la hora de traer los autorregistros completados a consulta para trabajar sobre ellos, quedando inservible el trabajo realizado durante el período entre sesiones.
- Dificultades del paciente para visualizar claramente los resultados obtenidos en los autorregistros en papel, así como para compararlos con resultados en semanas anteriores, para poder visualizar posibles progresos o empeoramientos.

Debido a estas razones, la posibilidad de desarrollar un sistema informatizado de diseño de autorregistros, envío y realización online en terminales móviles y visualización de resultados gráficos en consulta; se plantea como un paso esencial para poder mejorar el trabajo terapéutico.

Pese a ello, la gran mayoría del trabajo de programación de aplicaciones en el área de la psicología clínica ha ido más dirigido a favorecer exclusivamente el contacto por videoconferencia entre terapeuta y paciente, en lugar de profundizar en el uso de nuevas metodologías de diseño, registro y visualización de datos completados por el paciente de modo remoto.

1.2 Objetivos

Este trabajo se plantea desde el objetivo de poder facilitar una nueva herramienta a psicólogos clínicos y psiquiatras, con la que diseñar y ofrecer evaluaciones psicológicas online a sus pacientes para que éstos reciban una atención clínica personalizada tanto entre sesiones psicoterapéuticas como en seguimientos tras finalizar la terapia. Mediante la

nueva aplicación, los pacientes podrán llevar a cabo evaluaciones diseñados por sus terapeutas en los períodos entre sesiones, fuera de terapia, de un modo remoto, fácil y seguro.

Para llevar a cabo estos objetivos se han utilizado los siguientes recursos y herramientas de programación web:

- XAMPP: distribución de Apache que permite desplegar un servidor local en un ordenador personal. En él se han realizado las pruebas durante la programación de la página web.
- Se han utilizado lenguajes de programación web como HTML, CSS, JavaScript, PHP y de gestión de bases de datos como SQL.
- phpMyAdmin: software libre destinado a la gestión de bases de datos con la que se han configurado las bases de datos y tablas utilizadas en este proyecto.
- AWS: servicio de computación en la nube que ofrece, entre otros muchos productos, el servidor al que se ha subido la página web una vez se ha desarrollado el código necesario para que los usuarios de la Facultad de Psicología de la Universidad Complutense de Madrid empiecen a probar su funcionamiento.
- Bitnami: biblioteca de instaladores y paquetes de software para aplicaciones web. Es el servicio que se ha utilizado para desplegar el código de la página web en el servidor proporcionado por AWS.

Con todo esto se pretende desarrollar una aplicación web que proporcione recursos psicológicos efectivos de fácil acceso a los terapeutas y a sus pacientes, integrando nuevas tecnologías psicológicas para que sean usadas por los pacientes en su vida diaria.

La guía en dicho proceso de programación de la nueva herramienta vendrá de los futuros usuarios de este servicio: psicólogos clínicos del ámbito público y privado en la Comunidad de Madrid, que serán consultados por el equipo investigador e indicarán las necesidades específicas reales de evaluación remota que suelen encontrar en su práctica profesional y que deberían ser integradas en la nueva aplicación, de acuerdo con los principios técnicos y científicos comentados con anterioridad.

El proyecto, por tanto, supone un paso más allá en el diseño, puesta en funcionamiento y posible aceptación por los usuarios objetivo de un nuevo programa de evaluación psicológica online con amplio potencial para tener un alto impacto a nivel de uso.

2 Estado del arte

2.1 Introducción: autoinformes

Un autoinforme es un mensaje verbal que un individuo expresa sobre algún tipo de manifestación propia (Fernández-Ballesteros, 2011). Sin embargo, un autoinforme no solo puede darnos información sobre la experiencia subjetiva de un paciente como podría ser "estoy desanimado", sino que nos puede dar información objetiva, como el número de cigarrillos consumidos, o información externa al paciente, como la ocurrencia de un evento ajeno que ha desatado una respuesta en el paciente.

Los autoinformes presentan gran variabilidad en cuanto al contexto en el que son realizados, el tiempo y el tipo de preguntas y respuestas que pueden formar parte de él.

En cuanto al contexto, pueden ser realizados en situaciones artificiales, manipuladas dentro del laboratorio o consulta; o en situaciones naturales, donde el individuo registra la información sobre un suceso ocurrido en la vida real. Con relación al tiempo hay tres opciones. Pueden ser retrospectivos, que se refieren a hechos pasados; concurrentes, sobre sucesos que tienen lugar en el mismo momento en el que se informa, y, por último, futuros, que se refieren a sucesos que van a ocurrir.

En lo referente al tipo de preguntas, estas pueden ser de tipo estructuradas, semiestructuradas o no estructuradas. En el caso de las preguntas estructuradas, se tratan de preguntas claramente definidas que también deben ser respondidas de forma prefijada. En estas preguntas aparece información específica sobre el hecho por el que se pregunta. Un ejemplo de una pregunta de este tipo podría ser "¿Siente temblores al subir a un avión?". en la que se especifica tanto la situación en la que se produce, como la respuesta que provoca dicha situación en el sujeto. Por otro lado, las preguntas semiestructuradas tienen como objetivo que el individuo informe sobre el suceso con unas indicaciones mínimas, como puede ser ordenar situaciones que le provoquen temor de menor a mayor. Por último, las preguntas no estructuradas son aquellas en las que se da libertad al individuo a la hora de informar, con preguntas poco concretas, como "hable de usted mismo" o dejando que el paciente verbalice todos los pensamientos que se le ocurran.

Al existir diferentes tipos de preguntas también existen diferentes tipos de respuestas. El primer tipo es el de respuestas dicotómicas, por ejemplo, respuestas de sí o no. En estas se puede ofrecer la alternativa de no responder ninguna de las dos opciones para expresar duda o si no son aplicables. Otro tipo son las respuestas escalares, que sirven para determinar como de aplicable o no es una afirmación. Pueden utilizarse para ello escalas numéricas (del 1 al 10) o escalas verbales (nada, poco, algo, mucho). También existen las respuestas ordinales para que el individuo ordene elementos de manera jerárquica según sus preferencias. Para finalizar, el último tipo de respuesta es el de respuesta abierta, que se da cuando lo que se desea es conocer lo que realmente piensa el sujeto, por lo que se le permite responder de la forma que prefiera, con la mayor libertad posible.

Existen principalmente 4 tipos de autoinformes. El más extendido es la entrevista. En ella se solicita al individuo la información cara a cara. Estos datos pueden ser completados con más información proveniente de autobiografías o de la historia clínica del paciente. El segundo tipo es el de pensamientos en voz alta. Estos autoinformes suelen realizarse dentro del laboratorio o consulta, donde se recoge la verbalización a la vez que se realiza la conducta informada. Las conductas registradas suelen responder a estímulos proporcionados por el evaluador. Dentro de los pensamientos en voz alta, estos pueden ser de monólogo continuo (se verbalizan los pensamientos de forma simultánea a la realización de alguna tarea), muestras de pensamientos (solo se verbaliza cuando lo solicita el evaluador) y registro de eventos (se informa cuando tiene lugar alguna actividad interna de interés para el evaluador). El tercer tipo es el de cuestionarios, inventarios y escalas. En ellos se muestran preguntas a las que el individuo debe responder ante dos alternativas dicotómicas, elegir u ordenar una serie de elementos o mostrar el grado de adecuación del contenido de la pregunta. Por último, tenemos el autorregistro que vamos a ver a continuación (Fernández-Ballesteros, 2011).

2.2 Autorregistros

Cuando el individuo atiende a su propia conducta y la registra mediante un método establecido, estos autoinformes se denominan autorregistros. Las características principales de los autorregistros son las siguientes (Pérez Álvarez & Fernández-Ballesteros, 1994):

- Se suministra al paciente una hoja en la que se especifica la conducta que debe registrar y las condiciones para hacerlo. Siendo el mismo paciente el que completa el protocolo.
- Generalmente la conducta es registrada en el preciso momento en el que ocurre. En ocasiones se realiza tras ocurrir el evento, para evitar problemas de reactividad.
- Esta técnica es utilizada en situaciones reales.

A la hora de realizar los autorregistros el método más utilizado es el de lápiz y papel. Para ello es necesario que el terapeuta planifique dicho autorregistro. Debe seleccionar, y delimitar que conducta es la que se va a auto observar, las unidades de medida o parámetros con las que se va a medir (suele intentarse registrar con qué frecuencia ocurren las conductas o su duración, aunque también pueden medirse con parámetros cualitativos, como puede ser la intensidad o el sentimiento provocado en el paciente). Se debe diseñar el formato del protocolo de autorregistro, pudiendo incluir también los antecedentes o las consecuencias de la conducta que se esté estudiando.

Otro método propuesto para los autorregistros es el de usar contadores mecánicos, en los que el paciente pueda pulsar un botón para registrar la aparición de una conducta determinada. Así mismo se pueden utilizar dispositivos de tiempo, es decir, usar cronometro para registrar la duración de una conducta.

El último método posible para la realización de autorregistros es el de utilizar dispositivos electrónicos, como pueden ser una grabadora donde poder registrar información sobre la conducta que se está estudiando o un teléfono móvil.

Cigarrillo	Hora	Satisfacción	Situación
1			
2			
3			
4			

Ilustración 1: Ejemplo autorregistro sobre el consumo de cigarrillos.

Fecha	Situación	Emoción	Pensamientos
20-1-2020	Presentación delante de la clase	Miedo, nerviosismo	Lo voy a hacer mal, me voy a equivocar
22-1-2020	Examen de conducir	Angustia	Lo voy a suspender
2-2-2020	Cita médica	Incertidumbre	Van a detectar alguna anomalía

Ilustración 2: Ejemplo autorregistro situaciones que causan estrés.

2.2.1 Autorregistros en dispositivos móviles

Como hemos visto, los autorregistros son cuestionarios de distintos tipos que intentan recopilar información sobre las conductas que tienen relación con problemas psicológicos tales como los ya mencionados en la introducción (consumo problemático, sintomatología afectiva en interacciones sociales, etc.). Estos autorregistros son diseñados por un terapeuta y están destinados a que el paciente los complete durante el período entre sesiones. Generalmente, son realizados en papel, por lo que el paciente debe rellenarlos con papel y lápiz, pero como ya hemos visto, esta metodología presenta numerosos inconvenientes. Ante estos problemas surge la necesidad de realizar un servicio online que permita a los terapeutas diseñar estos autorregistros de manera sencilla y a los pacientes rellenarlos en cualquier momento y lugar a través de su smartphone.

Las evaluaciones psicológicas llevan tiempo intentando adaptarse al uso de internet. La aplicación de métodos online en el campo de la psicología proporciona ventajas en cuanto a coste y velocidad. Además, permite recoger y almacenar los datos obtenidos de

dichas evaluaciones de manera sencilla, al contrario de las evaluaciones realizadas en papel y lápiz (Naglieri, y otros, 2004).

Por tanto, hay numerosas aplicaciones destinadas a su utilización en el área de la psicología clínica, sin embargo, estas suelen estar orientadas a facilitar las videoconferencias entre el terapeuta y su paciente. En este primer grupo de aplicaciones, referidas a sistemas de videoconferencia, se pueden mencionar diferentes aplicaciones como, por ejemplo, AirPersons (www.airpersons.com/es), utilizada por todo tipo de profesionales (no solo psicólogos, si no también abogados o gestores, en la que se puede comprar una charla por videochat por un precio prefijado) o TherapyChat (<https://therapychat.com/>, en la que se asigna un psicólogo a cada usuario y se realizan las sesiones por videoconferencia en lugar de realizarlas de forma presencial). Este tipo de aplicaciones basan su utilidad en facilitar que las sesiones se realicen de forma telemática o facilitar el contacto entre un paciente y su psicólogo en el periodo entre sesiones, pero no tiene utilidad a la hora de proporcionar un medio en el que el paciente pueda apuntar información durante este periodo para que pueda ser analizado más tarde por el terapeuta como es el caso de los autorregistros.

Por otro lado, en el área de los autorregistros y su realización online, se cuenta con aplicaciones como Stop Panic & Anxiety Self-Help o Stresscheck. El primer caso se trata de una aplicación móvil de Excel at Life que cuenta con artículos destinados a ayudar al paciente, por ejemplo, a afrontar situaciones de crisis de pánico, así como audios de relajación y un diario en el que se pueden autorregistrar eventos y las emociones o efectos que provocan en el paciente. Esto podría ser utilizado a modo de autorregistros, pero cuenta con la desventaja de que los cuestionarios no pueden ser diseñados por el terapeuta y, por tanto, el paciente podría no registrar el evento con los detalles y la información relevante que el terapeuta desearía conocer. Por otro lado, Stresscheck permite realizar test predefinidos para puntuar el grado de estrés en un momento determinado y poder ver tanto su evolución a lo largo del tiempo, como su origen. Además, cuenta con ejercicios de relajación y yoga. En este caso, las pruebas tampoco pueden ser diseñadas por el terapeuta. Por tanto, este tipo de aplicaciones van dirigidas exclusivamente a que usuarios puedan monitorizar y registrar en sus móviles preocupaciones ocurridas durante su desempeño diario, sin un sistema de diseño específico de encuestas, o contacto con un segundo usuario receptor (terapeuta).

Aproximaciones más interesantes han sido recientemente desarrolladas, como el programa MentalCheck (<https://www.mentalcheck.es/>), el cual permite el diseño de encuestas por un determinado usuario (terapeuta), el envío de dichas encuestas a través de una aplicación en el teléfono móvil de un receptor (paciente) y la posibilidad de la visualización de resultados por parte de ambos en un momento posterior (consulta). Este tipo de diseño presenta claras ventajas respecto a las técnicas originales de autorregistro en lápiz y papel.

No obstante, los programas actuales presentan aún algunas limitaciones en cuanto a facilidad de uso y a su utilidad clínica real. Es por ello, que se plantea como necesario un nuevo proyecto que permita desarrollar una herramienta de autorregistros online que tenga en cuenta todas las necesidades específicas de los terapeutas y sus pacientes en la práctica real.

2.3 Aplicación web

A la hora de diseñar una página o aplicación web, existen numerosos lenguajes y opciones, así como portales que ofrecen servicios para poder diseñar una página web sin necesidad de programar o escribir código de ningún tipo, o con asistentes que te ayudan a diseñar tu web (Wordpress, Wix, IONOS). Sin embargo, para comenzar una página web desde cero, el sistema más sencillo es utilizar el entorno que nos proporciona XAMPP.

XAMPP es una distribución de Apache gratuita y fácil de instalar. Esta distribución contiene sistemas para la gestión de bases de datos MYSQL, scripts PHP y PERL. Es, además, una herramienta gratuita para su uso comercial o no comercial. Por tanto, mediante la instalación de XAMPP se puede comenzar la construcción de un sitio web en un entorno de desarrollo local, para después trasladarlo a la nube, en AWS, Microsoft Azure o Google Cloud Platform.

Por tanto, para realizar una página web utilizando XAMPP, la opción más sencilla es usar los lenguajes HTML y PHP.

2.3.1 HTML

HTML son las siglas de HyperText Markup Language, es decir, lenguaje de hipertexto de marcas. Este es el lenguaje básico para realizar páginas web. Es un lenguaje descriptivo, es el encargado de describir la página al navegador para que este pueda mostrarla. Así se le indican los elementos que debe mostrar y como mostrarlos, por ejemplo, donde comienza un párrafo, donde se muestra una lista, etc.

Los comandos en este lenguaje son llamados tags o etiquetas, y se deben escribir entre los símbolos < >. Todos los documentos HTML deben comenzar con la etiqueta <html> y terminar con </html> y se dividen en dos bloques, la cabecera, donde se incluye información que concierne a toda la página (como el título), y el cuerpo, que supone el resto de la página.

Un elemento muy importante del lenguaje HTML son sus formularios. Estos están delimitados por las etiquetas <form> y </form> y son elementos que nos permiten solicitar datos al usuario y que este pueda introducirlos en nuestra página web para enviárnoslos. Para que estos datos puedan ser recibidos se debe utilizar un CGI (Common Gateway Interface o interfaz de entrada común), es decir un programa que se ejecute en el lado del servidor. Al enviar un formulario al servidor de nuestra web se procesará la información y se devolverá una respuesta. Este envío de datos se realiza indicando el atributo METHOD, que puede tomar los valores GET y POST (Ferrer , García , & García).

2.3.2 Métodos GET y POST

Los datos introducidos en el formulario se envían al agente identificado en el atributo “action”.

- Si el método elegido es GET, y el atributo “action” es una HTTP URI (Identificador de recursos uniforme, es decir, donde se van a procesar los datos introducidos en el formulario), el cliente añade un “?” y después añade los pares de nombre y valor de los datos introducidos al enviar el formulario.
- Si, por el contrario, el método elegido es POST y el atributo “action” es una HTTP URI, el cliente envía una transacción HTTP POST con los datos introducidos.

El método GET suele usarse cuando se van a realizar acciones que no causan ningún efecto. por ejemplo, realizar una búsqueda en base de datos. En cambio, el método POST se suele utilizar cuando las acciones que se van a realizar si causan algún efecto, como puede ser, realizar un cambio en algún registro de una base de datos (Form submission. HTML 4.01 Specification. W3C, 1999).

En el caso de que un formulario sirva para registrar el nombre y apellido de una persona y esos datos deban ser tratados en el archivo “action.php”, si se usara GET, los datos serían enviados en la URI de la siguiente forma:

action.php?nombre=José&apellido=García.

En cambio, si el método elegido fuera POST, los datos serían enviados en una solicitud POST HTTP de esta forma:

nombre=José&apellido=García

2.3.3 PHP

PHP fue creado en 1995 por Rasmus Lerdorf, que diseñó un script para poder comprobar cuanta gente leía su currículum online y poder mostrar un contador en él. Este conjunto de herramientas fue denominado "Personal home page", de ahí el nombre PHP. Este hecho suscitó bastante interés, por lo que Lerdorf siguió desarrollando PHP hasta el lanzamiento de PHP 2.0 en 1997 con numerosos instrumentos para la programación web (Jason Gilmore, 2010), hasta su versión actual, PHP 7.4.

La principal característica de PHP es que está diseñado para realizar tareas en el lado del servidor. El código se ejecuta en el lado del servidor, donde se genera HTML que se envía al cliente. De este modo, se pueden enviar datos mediante los formularios de HTML al servidor que serán procesados con PHP.

2.3.4 MySQL

MySQL es un sistema de gestión de bases de datos. Una base de datos es una colección de datos estructurados, y dado que en nuestra web vamos a tener datos sobre los usuarios de los autorregistros, debemos utilizar este sistema para acceder, añadir y procesar

estos datos. Las bases de datos de MySQL son relacionales, es decir, la información se almacena en tablas separadas. Además, la estructura está organizada en archivos optimizados para lograr mayor velocidad y su modelo lógico, con tablas, filas y columnas permite un entorno de programación flexible.

En este sistema se utiliza el lenguaje SQL (structured query language), el lenguaje más utilizado en la gestión de bases de datos.

MySQL es un software Open Source, por lo que se puede descargar de manera gratuita. Además, los servidores MySQL pueden correr en todo tipo de dispositivos, desde un portátil u ordenador personal a un servidor web. También es fácilmente escalable y se pueden llegar a usar bases de datos con 50.000.000 de registros, habiendo casos de servidores con 200.000 tablas y 5.000.000.000 columnas. MySQL cuenta con APIS para C, C++, Eiffel, Java, Perl, PHP, Python, Ruby y TCL (Overview of the MySQL Database Management System, s.f.).

2.3.5 phpMyAdmin

PhpMyAdmin es un software libre escrito en PHP y utilizado para la administración de MySQL desde nuestro navegador web. Al instalar y utilizar XAMPP podemos utilizar phpMyAdmin para gestionar las bases de datos de MySQL con una interfaz de usuario intuitiva, así como ejecutando directamente sentencias SQL (Bringing MySQL to the web, s.f.).

2.3.6 CSS

CSS (Cascading Style Sheet u hojas de estilo en cascada) es un lenguaje de diseño gráfico. Este lenguaje proporciona amplias posibilidades en el diseño y representación de las páginas web y se utiliza para fijar las características de formato de los elementos HTML. Fue definido en 1996, y nos ofrece la posibilidad de vincular los documentos HTML con hojas de estilo que contienen la información topográfica de los componentes visuales de una página y separar la estructura de los contenidos de su apariencia. Además, CSS permite ajustar la visualización de la página a todo tipo de pantallas y dispositivos, desde smartphones a monitores de ordenador. El uso de CSS tiene por tanto claras ventajas, ya que se permite mantener el mismo estilo en diferentes páginas y separar el contenido de su visualización.

La incorporación de CSS al HTML de una página se puede realizar de tres formas. La primera, añadiendo instrucciones del estilo directamente en la etiqueta correspondiente del archivo HTML. La segunda, incorporando las instrucciones CSS en el elemento `<style>`, con el atributo `“type=text/css”`. La última opción es definir todas las instrucciones en un archivo CSS externo que será referenciado en todos los documentos HTML en el que se vaya a aplicar el estilo que se ha definido en dicho CSS (Schulz, 2008).

2.3.7 JavaScript

JavaScript es un lenguaje de programación utilizado para diseñar páginas webs dinámicas. Con esto nos referimos a páginas en las que la información que se muestra en ella puede cambiar, dependiendo de las acciones que realice el usuario, por ejemplo, que salga un mensaje de aviso cada vez que este pulse un botón. JavaScript es un lenguaje interpretado, es decir, no es necesario que el código sea compilado antes de ser ejecutado.

Este lenguaje surgió en los años 90, cuando la conexión a internet era de muy baja velocidad. Debido a ello apareció la necesidad de que un lenguaje de programación permitiera la ejecución del lado del navegador o cliente, sin necesidad de tener que esperar la respuesta del servidor cada vez que el usuario realizaba alguna acción en la página. Por ejemplo, a la hora de rellenar un formulario, si el usuario lo hacía de forma incorrecta, poder indicárselo sin tener que enviar los datos introducidos al servidor y esperar la respuesta de este para notificar el error al usuario. Por esta razón Brendan Eich, programador de Netscape desarrollo un primer lenguaje adaptando las tecnologías existentes al navegador Netscape Navigator. Este primer lenguaje se denominó LiveScript. Más tarde se cambiaría el nombre a JavaScript y en 1997 se envió su especificación al ECMA (European Computer Manufacturers Association) (Eguíluz Pérez, 2009).

El código JavaScript puede integrarse con HTML en dos modos distintos. La primera opción es incorporar el código entre las etiquetas `<script>`. La segunda opción es incluir el código JavaScript en un archivo externo al que se haga referencia desde el archivo HTML.

3 Diseño y desarrollo

3.1 *Objetivos de diseño*

El objetivo de este trabajo de fin de máster es realizar una aplicación web que permita el uso de autorregistros para terapeutas y sus pacientes. Por un lado, deberá permitir a los terapeutas crear los autorregistros con diferentes tipos de preguntas y asignárselos a los pacientes correspondientes. Por otro lado, los pacientes deben poder acceder a los autorregistros diseñados por sus terapeutas y rellenarlos en el momento que deseen o que se les haya asignado y poder enviar sus respuestas a sus terapeutas. Además, se deben almacenar los resultados de cada autorregistro para poder ser visualizados por el terapeuta una vez hayan sido completados por el paciente.

Por tanto, uno de los aspectos más importantes es la facilidad de uso. La página web a la que accedan tanto los terapeutas como los pacientes debe ser intuitiva y sencilla de usar, que resulte cómoda incluso para usuarios que no estén demasiado familiarizados con internet. Para ello se han seguido las recomendaciones especificadas por el profesor Álvaro Sánchez López (Grupo de Investigación Consolidado en Psicopatología y Salud Mental Positiva de la Universidad Complutense de Madrid).

3.1.1 Diseño de la aplicación web

Antes de comenzar a construir la aplicación es importante definir la estructura que tendrá, que páginas y opciones se mostraran a los usuarios. Comenzaremos con la aplicación destinada a los terapeutas. La estructura que tendrá es la siguiente:

En primer lugar, se mostrará una pantalla de inicio de sesión o registro. Tras introducir los datos de usuario y contraseña se accede a una pantalla principal donde se muestran las 4 opciones principales. La primera es la sección de crear autorregistros, donde se especificarán las preguntas (así como el tipo de preguntas, pudiendo elegir también las opciones de respuesta que se dan al paciente) que formarán un autorregistro. Tras introducir las preguntas se podrá guardar ese autorregistro asignándole un nombre identificador. La segunda opción es mostrar la lista de pacientes. En ella se mostrarán todos los pacientes que tenga asignados el terapeuta, además de una opción para agregar nuevos pacientes mediante su nombre y su correo electrónico. La tercera opción es la de mostrar la lista de autorregistros. En ella aparecerán todos los autorregistros creados y guardados por el terapeuta con la opción de asignárselos al paciente que debe rellenarlos. Por último, se muestra la opción de mostrar resultados, donde se podrá acceder a las respuestas completadas por el paciente para cada autorregistro.

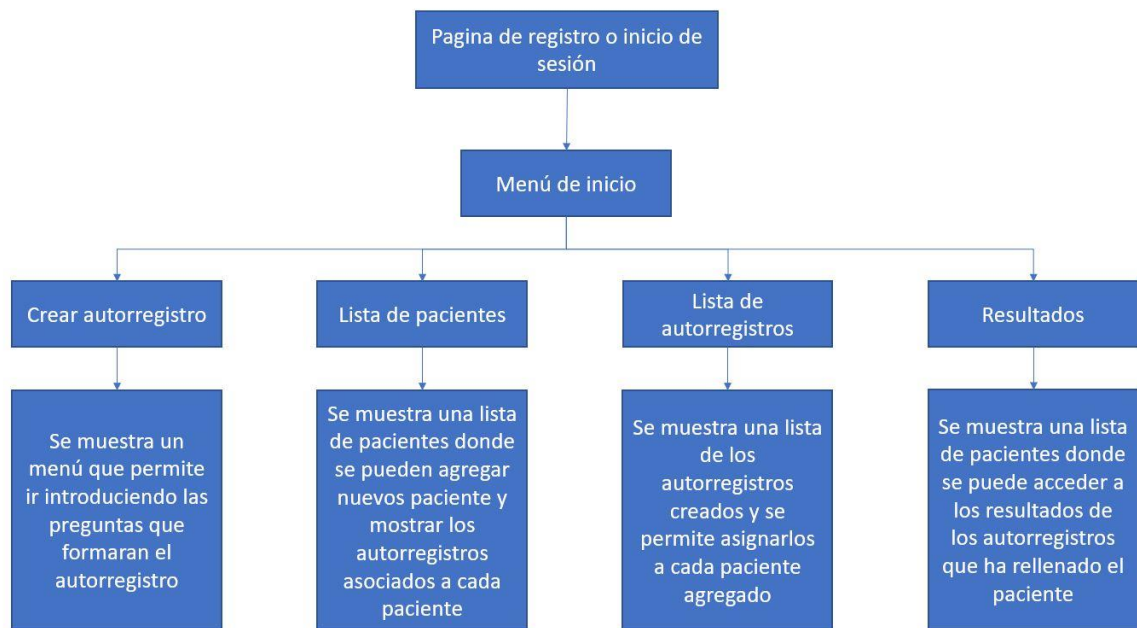


Ilustración 3: Estructura página web destinada a los terapeutas.

En el caso de la aplicación destinada a los pacientes, la estructura es más sencilla. En este caso solo se mostrará una pantalla de inicio de sesión y registro desde la cual podrán acceder a su perfil, donde se encuentra una lista de los autorregistros asignados disponibles para ser completados. Pulsando en cada uno podrán acceder a ellos para completarlos y enviar sus respuestas que quedaran almacenadas en nuestra base de datos.



Ilustración 4: Estructura página web destinada a los pacientes.

En ambas páginas, tanto en la destinada a los terapeutas como la destinada a los pacientes se mostrará también la opción de editar el perfil, así como de borrarlo.

3.2 Primeros pasos

Una vez hemos diseñado la estructura de la aplicación web que vamos a construir, el primer paso que debemos realizar es instalar XAMPP.

3.2.1 Instalación de XAMPP

El proceso de instalación de XAMPP es bastante sencillo. Solo es necesario descargar el instalador desde su página web oficial (<https://www.apachefriends.org/es/index.html>), donde además contamos con un vídeo explicativo que nos guía paso a paso. Este instalador nos permite seleccionar los componentes que deseamos instalar (MySQL, PHP, phpMyAdmin, etc. en nuestro caso instalaremos todos) así como el directorio donde lo queremos colocar. Tras una breve espera la instalación está completada y podemos acceder al panel de control de XAMPP.

Este panel de control está disponible bajo el nombre “xampp_control” en la carpeta “xampp”. Al acceder a él podemos seleccionar los módulos que queremos ejecutar.

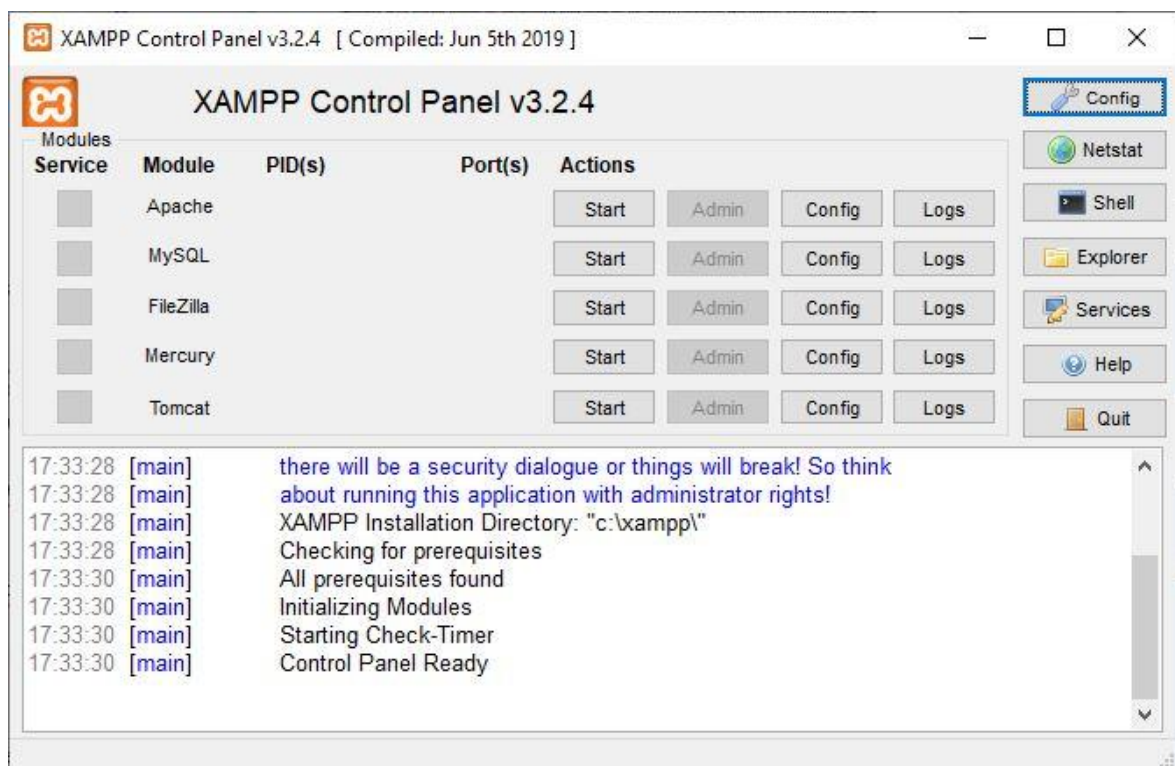


Ilustración 5 : Panel de control de XAMPP.

Tras realizar esta instalación ya podemos comenzar a escribir nuestro código para poner en funcionamiento tanto la página web destinada a los terapeutas como a los pacientes. Este código debe ser emplazado en la carpeta “htdocs”, dentro de la carpeta de

“xampp”. Sin embargo, mientras usemos XAMPP, solo podremos acceder a la página web que estemos creando mientras tengamos los módulos correspondientes de XAMPP ejecutándose, es decir, la utilización de XAMPP está destinada a crear la web y realizar las pruebas en nuestro ordenador personal, la web no estará disponible hasta que no subamos el código definitivo a un servidor online.

Tras esto ya podremos acceder a las páginas web que creemos en nuestro servidor local, así como a la interfaz de usuario de phpMyAdmin desde donde podremos gestionar las bases de datos a las que accederá nuestra página web. Si por ejemplo creamos una página web con el nombre index.html podremos acceder a ella iniciando los módulos de Apache y MySQL en el panel de control de XAMPP e introduciendo la dirección `http://localhost/index.html` en el navegador.

3.2.2 Configuración de phpMyAdmin

Para acceder a la interfaz de phpMyAdmin y comenzar a crear y gestionar las bases de datos de nuestra página deberemos introducir la dirección `http://localhost/phpmyadmin/` en el navegador.

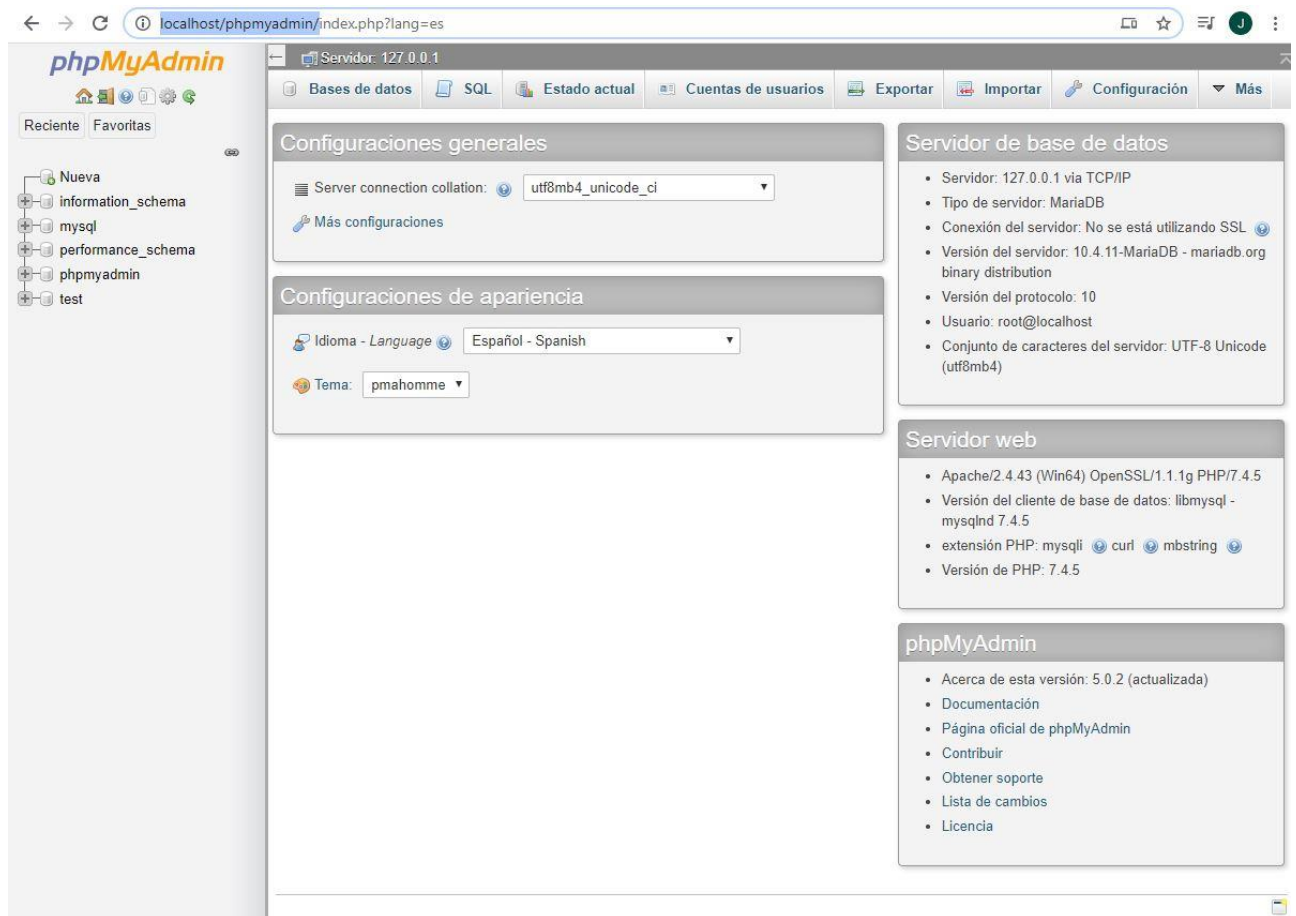


Ilustración 6: Interfaz de phpMyAdmin.

Una vez accedemos a la interfaz de phpMyAdmin debemos crear las bases de datos y tablas con las que vamos a trabajar en nuestra aplicación web. Las tablas que crearemos para almacenar la información sobre los usuarios y los autorregistros serán las siguientes:

- En primer lugar, crearemos las tablas que contendrán los usuarios, una tabla para pacientes y otra para terapeutas, donde se almacenarán los nombres, correos electrónicos y contraseñas de todos los usuarios que se registren en nuestra web.
- La siguiente tabla que crearemos será la de autorregistros, en ella se almacenarán el nombre asignado a cada autorregistro y el terapeuta que lo ha creado (y por tanto el único que puede acceder a él).
- La tercera tabla creada es la tabla que almacenará las preguntas de cada autorregistro. En ella se almacenará la pregunta, el tipo de respuesta a dicha pregunta (respuesta libre, respuesta numérica, respuesta con escala verbal de opciones o respuesta de escala numérica), y el cuestionario al que está asociada dicha pregunta. Además, se creará una tabla adicional con estos mismos datos que servirá para guardar las preguntas que aún no hayan sido asignadas a ningún cuestionario.
- Por último, cada vez que se registre un paciente se creará una tabla propia para cada uno en la que se almacenarán los autorregistros que se le han asignado, así como la fecha en la que debe completarlos, y una tabla donde se almacenarán las respuestas de los autorregistros que el paciente ha completado.

3.2.3 Formularios HTML y acceso a la base de datos

El funcionamiento de la página web de este proyecto va a estar basado en el uso de los formularios HTML. Estos nos proporcionan una herramienta muy útil y sencilla para que un usuario envíe información a nuestro servidor. Como hemos dicho, estos formularios están delimitados por las etiquetas `<form>` y `</form>`.

Dentro del formulario es necesario especificar los campos “action”, que define el archivo con el código en el que se procesarán los datos enviados en el formulario, y el método (GET o POST, como ya se ha indicado en la sección de estado del arte). También existen numerosas opciones dependiendo del tipo de datos que se quiere obtener en el formulario. Estos se denominan “input types” y varían, por ejemplo, para indicar si en el formulario el usuario debe introducir un texto, un número, una contraseña (quedando esta oculta en la pantalla del dispositivo), una elección dentro de una lista, etc.

Un ejemplo del código necesario para realizar un formulario sencillo en el que se pide al usuario que introduzca su nombre y lo envíe sería el siguiente:

```
<form action="action.php" method="post">
  Nombre:
  <input type="text" name="nombre"><br>
  <input type="submit" value="Envíar">
</form>
```

Este código tendría en el navegador el siguiente aspecto:



Nombre:

Enviar

Ilustración 7: Ejemplo aspecto formulario HTML.

Al introducir un nombre como “Juan” y pulsar en enviar, en este caso se enviaría un POST al servidor en el que el campo “nombre” tendría el valor de “Juan”. De este modo, en el archivo “action.php” se podría recoger ese dato mediante la instrucción “\$nombre_recibido=\$_POST[“nombre”];” para procesarlo, quedando el valor “Juan” guardado en la variable “\$nombre_recibido”.

Una vez recibido el dato, en este caso, el nombre, se podría almacenar en nuestras bases de datos. Para ello es necesario realizar la conexión a la base de datos en el archivo PHP y ejecutar la consulta para guardar dicho valor. Esto se realiza mediante la función “mysqli_connect” que recibe como parámetros el nombre del host o su dirección IP, el usuario de MySQL, la contraseña de dicho usuario, y el nombre de la base de datos a la que queremos acceder. Esta función devuelve un objeto que sería la conexión al servidor MySQL (Documentación mysqli_connect, s.f.). Este objeto es necesario pasarlo como argumento a la función “mysqli_query”, junto con una cadena en la que tengamos almacenada la consulta que vamos a ejecutar. La función “mysqli_query” ejecuta la consulta en la base de datos y devuelve FALSE en caso de error o un objeto con el resultado (Documentación mysqli_query, s.f.). Una vez obtenido el resultado, este se puede procesar para mostrarlo o realizar las acciones correspondientes a la petición realizada por el usuario de la web.

Es importante que a la hora de acceder a la base de datos tengamos cuidado y evitemos el denominado “sql injection”. Esto podría ocurrir si, por ejemplo, en el caso del formulario explicado anteriormente, un usuario introdujera el valor “or 1=1”. De este modo, un select sobre una tabla de usuarios podría quedar de la siguiente manera “SELECT * FROM tabla_usuarios WHERE nombre= “” or 1=1”; lo que devolvería todos los registros de la tabla usuarios. Una función útil para evitarlo es “mysqli_real_escape_string”, que recibe como parámetro una cadena de la que se eliminan los caracteres NULL (ASCII 0), \n, \r, \, ', ", y Control-Z, para evitar estas situaciones de riesgo (Documentación mysqli_real_escape_string, s.f.).

3.2.4 Desarrollos formularios de registro e inicio de sesión

Para comenzar a construir la página web, el primer paso fue realizar un formulario que permitiera el registro de los usuarios y el inicio de sesión. Estos formularios son similares para el caso de la web destinada a pacientes y la web destinada a terapeutas.

En el caso del registro, se piden los datos nombre, correo, contraseña y repetición de contraseña. Se envían por método POST al servidor y a continuación se comprueba si el correo ya está utilizado por otro usuario. Si el correo está utilizado por otro usuario se muestra un mensaje indicándolo y se le redirige al formulario de inicio de sesión. Si la

contraseña no coincide con la repetición de la contraseña, se muestra un mensaje indicándolo y se redirige de nuevo al formulario de registro. En cambio, si los datos son correctos, es decir, si el correo no está en uso por ningún usuario y la contraseña y la repetición de la contraseña coinciden, se crea el usuario actualizando las tablas respectivas con los datos del usuario. Si se trata de un paciente, se crearán las tablas asociadas a dicho paciente que contendrán los autorregistros que se le han asignado y las respuestas que ha enviado.

Registro

Nombre

Email

Contraseña

Repetir contraseña

[Registrarse](#)

Ilustración 8: Formulario registro pacientes.

En el caso de inicio de sesión se solicita el correo y la contraseña. Siguiendo el mismo proceso se comprueba si el usuario está registrado, en cuyo caso se redirige al menú principal de la aplicación y se establece un identificador de sesión con la variable superglobal `$_SESSION`. Esta variable será accesible tras ejecutar la función “`session_start`” en PHP (Documentación uso básico `$_SESSION`, s.f.). En caso de que el email o la contraseña no coincida con ninguno de los usuarios almacenados en la base de datos se muestra un mensaje indicando que el usuario o la contraseña son incorrectos y se redirige al formulario de inicio de sesión.

Inicio de sesión

Email

Contraseña

[Iniciar sesión](#)

Ilustración 9: Formulario inicio de sesión pacientes.

Después de que el usuario se haya registrado e iniciado sesión, las páginas que se mostrarán depende de si son terapeutas o si son pacientes.

3.3 Desarrollo web destinada a terapeutas

En el caso de los terapeutas, cuando se ha iniciado sesión se muestran las cuatro opciones principales que puede realizar el terapeuta: crear autorregistro, lista de pacientes, lista de autorregistros y resultados.

3.3.1 Crear autorregistros

En el caso de la opción de crear autorregistro se mostrará una pantalla en la que se podrán introducir las preguntas que formarán parte del autorregistro.



The screenshot shows a web interface for creating a self-registration. At the top is a dark navigation bar with five links: 'Crear autorregistro' (highlighted in green), 'Lista de autorregistro', 'Lista de pacientes', 'Resultados', and 'Perfil'. On the far right of this bar is a link 'Cerrar sesión'. Below the navigation bar, the main heading is 'Inserte una pregunta'. Underneath is a large, empty text input box. Below the input box is the label 'Escoja el tipo de respuesta a la pregunta', followed by four radio button options: 'Escala verbal', 'Escala numérica', 'Número', and 'Respuesta libre'. At the bottom of the form is a green button labeled 'Añadir pregunta'.

Ilustración 10: Creación de autorregistros

Como se puede observar en la ilustración 10, en primer lugar, se muestra una barra de navegación en la parte superior de la pantalla, para facilitar la navegación por las opciones que permite la página. Más abajo se muestra un cuadro de inserción de texto donde el terapeuta debe introducir la pregunta, un cuadro de elección del tipo de respuesta y un botón para añadir la pregunta al autorregistro. Este cuadro de inserción de texto está construido como un formulario. En él se envían con el método POST la pregunta introducida y el tipo de respuesta escogido, y el identificador del terapeuta que la ha redactado. Al recibirse estos datos en el servidor, se almacenan en la base de datos como preguntas temporales, sin asociarse a ningún autorregistro todavía.

Dependiendo del tipo de respuesta que se escoja, al pulsar añadir pregunta podrán aparecer varias opciones. Si se escoge el tipo 'Escala verbal' aparecerá un mensaje indicando cuantas opciones se desean introducir y tras esto se podrán introducir tantas opciones como se hayan indicado. De esta forma se añaden la pregunta y las opciones que le aparecerán al paciente cuando vaya a rellenar el autorregistro. Si se escoge la opción 'Escala numérica', se mostrará un mensaje para que introduzcamos los valores superior e inferior entre los que podrá elegir el paciente cuando vaya a rellenar el autorregistro.

Finalmente, si se escogen las opciones 'Número' o 'Respuesta libre', las preguntas se añadirán directamente al autorregistro que estamos creando. Tras introducir cada pregunta, en el código PHP de esta página se realizan consultas que mostrarán en la pantalla las preguntas que se han ido almacenando en la base de datos como temporales, sin ser asociadas aún a ningún autorregistro para que el terapeuta se haga una idea de cómo quedará el autorregistro al que tendrá acceso el paciente. Por último, se muestra un botón para guardar el autorregistro que se está creando.

En la ilustración 11 se puede observar un ejemplo de un autorregistro creado para controlar el consumo de cigarrillos. Debajo de la opción 'Inserte una pregunta' se van mostrando las preguntas que se han añadido para que el terapeuta vea el resultado preliminar del autorregistro que más tarde podrá asignar al paciente que desee. La primera pregunta es de tipo 'Número', la segunda, de tipo 'Escala verbal', donde se han introducido dos opciones como posibles respuestas del paciente, y la tercera de tipo 'Escala numérica', donde el paciente podrá mover una escala para seleccionar el valor que desee. Además, se da la opción de borrar cada pregunta pulsando el botón 'Eliminar pregunta'. Al pulsar en el botón de eliminar pregunta se envía también por POST el id de la pregunta que se desea eliminar, de modo que si se recibe un POST para eliminar se ejecute la consulta DELETE de la tabla con las preguntas almacenadas como temporales. En la parte inferior de la página se muestra el botón 'Guardar autorregistro'.

[Crear autorregistro](#) [Lista de autorregistro](#) [Lista de pacientes](#) [Resultados](#) [Perfil](#) [Cerrar sesión](#)

Inserte una pregunta

Escoja el tipo de respuesta a la pregunta

☐ Escala verbal
☐ Escala numérica
☐ Número
☐ Respuesta libre

[Añadir pregunta](#)

Autorregistro

¿Cuántos cigarrillos ha fumado hoy?

[Eliminar pregunta](#)

¿En que situación los ha fumado?
☐ En casa
☐ En el trabajo
[Eliminar pregunta](#)

¿Cuanto placer ha obtenido fumandolos?
1 100
[Eliminar pregunta](#)

[Guardar autorregistro](#)

Ilustración 11: Creación autorregistro.

Cuando el terapeuta haya finalizado la creación del autorregistro y pulsado el botón 'Guardar autorregistro', se mostrará un mensaje para solicitar que el terapeuta asigne un nombre al autorregistro. Tras introducirlo, se envía un POST con el nombre elegido y todas las preguntas redactadas por el terapeuta que estaban guardadas en la tabla como temporales se trasladan a la tabla con las preguntas definitivas y se les asigna un identificador del autorregistro al que pertenecen. El autorregistro guardado quedará disponible en la opción 'Lista de autorregistros'.

3.3.2 Lista de autorregistros

La segunda opción de la barra de navegación es la de lista de autorregistros. Al acceder a ella se muestran todos los autorregistros que ha diseñado el terapeuta. Se obtienen mediante una consulta a la tabla que contiene los autorregistros mostrando solo los que se corresponden con el terapeuta identificado en la variable `$_SESSION`, y devolviendo su identificador y nombre. Se muestra el nombre con el que se ha guardado cada autorregistro y se da la opción de abrir el autorregistro para consultar las preguntas de las que consta, enviar el autorregistro a un paciente y eliminar el autorregistro.



Ilustración 12: Lista de autorregistros.

En caso de elegir la opción de enviar un autorregistro se mostrarán todos los pacientes que el terapeuta tiene agregados. Tras elegir el paciente al que se desea enviar, se podrá elegir la fecha o la periodicidad con que debe completar el paciente el autorregistro. Se podrá elegir si el paciente debe rellenar el autorregistro en una fecha concreta, si lo debe rellenar con periodicidad semanal (indicando que días de la semana lo debe completar), con periodicidad mensual (indicando el día del mes que se debe enviar) o si el paciente puede rellenar el autorregistro en cualquier momento, cuando el propio paciente lo decida.

Crear autorregistro
Lista de autorregistro
Lista de pacientes
Resultados
Perfil
Cerrar sesión

Escoja cuando debe completar el autorregistro el paciente

Si desea que el paciente solo responda una vez al autorregistro inserte una fecha

Si desea que el paciente solo responda semanalmente al autorregistro inserte que días debe completarlo

☐ Lunes
☐ Martes
☐ Miércoles
☐ Jueves
☐ Viernes
☐ Sabados
☐ Domingos

Si desea que el paciente solo responda mensualmente al autorregistro inserte que día de cada mes debe completarlo

Si desea que el paciente reciba correo electronicos recordatorios cuando deba completar el autoregistro, marque esta casilla

☐ Recordatorio

Si desea que el paciente pueda completar el autoregistro en cualquier momento tantas veces como quiera, marque esta casilla

☐ Libre

Guardar

Ilustración 13: Selección de fechas.

3.3.3 Lista de pacientes

En esta pestaña se mostrarán los pacientes que cada terapeuta ha agregado. Al igual que en el caso de la lista de autorregistros, esta lista de pacientes se obtiene mediante la consulta en la que se busca en la tabla de usuarios los pacientes que tiene asignado el terapeuta identificado en la variable `$_SESSION`. Además, se mostrará una opción para agregar a un paciente. Para ello, el paciente debe indicar a su terapeuta el correo y el nombre con el que se ha registrado. A continuación, el terapeuta debe introducir los datos en esta pestaña. Se envía un POST con estos datos y si no coinciden con ningún paciente registrado se mostrará un mensaje indicándolo y pidiendo los datos de nuevo. Si los datos son correctos, el paciente será agregado con éxito. Además, si el terapeuta pulsa sobre alguno de los pacientes, se enviará el identificador del paciente seleccionado al servidor y se devolverá la lista de los autorregistros que han sido asignados a cada paciente. En la ilustración 14 se muestra un ejemplo de la opción añadir paciente y una lista de un terapeuta que tiene un solo paciente agregado.

Crear autorregistro
Lista de autorregistros
Lista de pacientes
Resultados
Perfil
Cerrar sesión

Añadir nuevo paciente

Lista de pacientes

Nombre: Juan, correo:aaaaaaaaaaaaa@aaaaa.com

Ilustración 14: Lista de pacientes.

3.3.4 Resultados

En la pestaña de resultados se muestran las respuestas que cada paciente ha dado a un autorregistro. Para ello se mostrará el listado de pacientes de modo que al pulsar sobre uno de ellos accedamos a los autorregistros que tiene asignados este paciente, y tras esto, seleccionando uno de los autorregistros podemos observar las respuestas que el paciente ha dado al autorregistro.

Una parte muy importante sobre la relevancia de este proyecto era crear la plataforma web de la manera más amigable para el usuario, de forma que fuera fácil de entender y usar. Por ello, era imprescindible contar con una opción donde los terapeutas pudieran observar y analizar los resultados de los autorregistros realizados por sus pacientes de forma clara y visual. De este modo podrían acceder a la página durante la consulta con su paciente y mostrarle los resultados para comentarlos y estudiarlos con él, interpretando los resultados con un vistazo rápido.

Con este fin se propuso que se pudieran ver los resultados en diferentes formatos, como podrían ser gráficos de barras, histogramas o tablas de texto, como los ejemplos indicados en la ilustración 15. En ella se observa cómo debería mostrarse la respuesta a una pregunta en un autorregistro para controlar el consumo de cigarrillos, mostrando cuantas veces ha respondido el paciente si ha fumado en casa, en el trabajo o de paseo. En esta figura se muestra también un ejemplo de respuestas a un autorregistro que tenga que ver con el control del uso de redes sociales, indicando la fecha, la duración y el tiempo total de las conexiones.

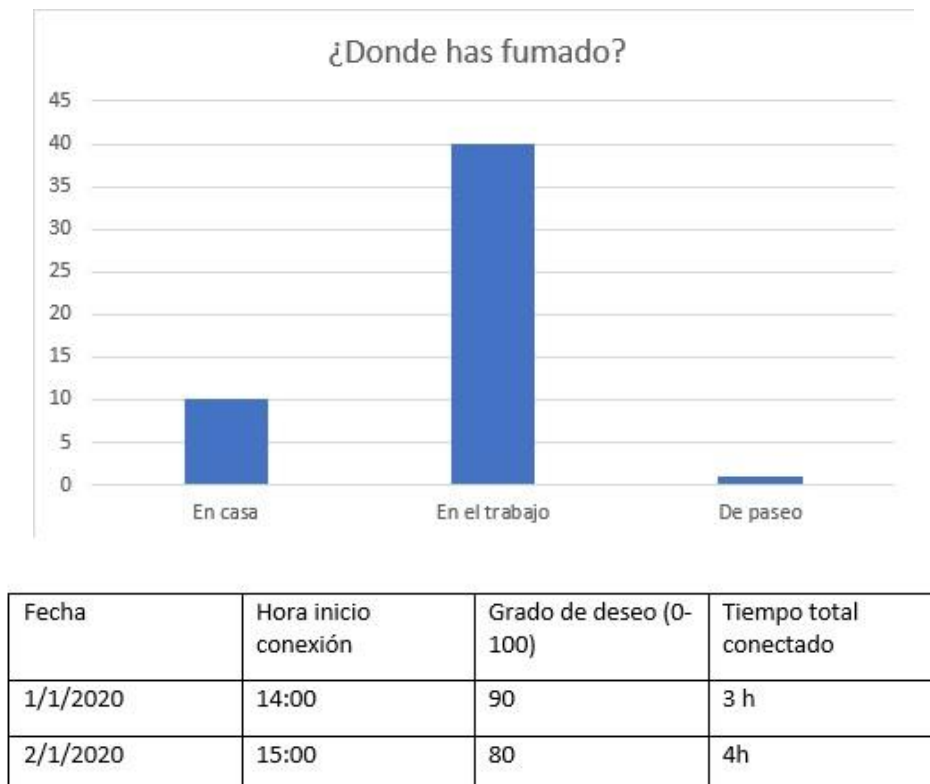


Ilustración 15: Ejemplo resultados.

Para mostrar los resultados de las respuestas a los autorregistros realizados por el paciente se utilizó la herramienta Google Charts. Esta es una utilidad de Google de código abierto que permite visualizar estos datos de forma sencilla, tanto en dispositivos Android como iPhones y en todos los navegadores, sin necesidad de instalar ningún plugin.

Además, Google Charts cuenta con numerosos tipos de gráficos, desde gráficos de barras o histogramas hasta gráficos geográficos y todos ellos cuentan con ejemplos de uso disponibles en <https://developers-dot-devsite-v2-prod.appspot.com/chart/interactive/docs>. Para añadir gráficos a nuestra página web solo es necesario cargar las librerías usadas por esta herramienta desde <https://www.gstatic.com/charts/loader.js>. Para ello se debe usar JavaScript, que se puede insertar en el código HTML de nuestra página web. De este modo insertamos los datos que deseamos mostrar en nuestro gráfico en el código JavaScript, seleccionamos las opciones de nuestro gráfico y así creamos un objeto con el id que deseamos que será mostrado en nuestra página web.



Ilustración 16: Ejemplo página resultados.

En la ilustración 16 se muestra un ejemplo de cómo se podrían visualizar las respuestas que ha ido enviando un paciente a un autorregistro que se le haya asignado.

3.3.5 Perfil

La última opción que tiene el usuario es la de 'Perfil'. En esta pestaña se encuentra la información personal del usuario, es decir, el correo y la contraseña. En ella se da la opción de cambiar tanto uno como otro, así como una opción para eliminar toda la información del perfil. Si se pulsa el botón eliminar se mostrará un mensaje de advertencia en el que se indica que se borrará toda la información de este usuario en nuestra base de datos, además, los autorregistros creados por este terapeuta serán también borrados y los pacientes que tenga asignados no podrán acceder a dichos autorregistros. Tras pulsar esta opción y confirmar que el usuario desea borrar estos datos se cierra la sesión y se redirige a la página principal de inicio de sesión y registro.

3.3.6 Cerrar sesión

La última opción de la barra superior de navegación es la de 'Cerrar sesión'. Al pulsar esta opción se destruye la sesión utilizando la función “`session_destroy`” y se redirige a la página de inicio de sesión.

3.4 Desarrollo web destinada a pacientes

En el caso de la web destinada a pacientes tenemos una interfaz igual que la de la página destinada a los terapeutas. La diferencia es que la web destinada a los pacientes es más simple y cuenta con menos opciones. También cuenta con una barra de navegación superior como en el caso de la página de terapeutas, pero en este caso solo tenemos tres opciones.

La primera opción es la de 'Menú'. En esta opción se muestran los autorregistros que han sido asignados al paciente. En caso de que el paciente no le haya comunicado su correo electrónico y el nombre con el que se ha registrado en la base de datos a su terapeuta, este aún no habrá podido agregar al paciente por lo que no se mostrará ningún autorregistro que completar.

En el momento en que el terapeuta agregue al paciente introduciendo sus datos y asignándole un autorregistro, aparecerá una lista con todos los autorregistros que debe completar el paciente, así como con las fechas en que debe hacerlo.

El funcionamiento interno de esta página es similar al de la página de los terapeutas. Al iniciar sesión también se establece un identificador de sesión para el paciente en la variable `$_SESSION`. Tras esto se muestran los cuestionarios asociados al paciente y se disponen como distintos formularios en los que al pulsar en la opción “abrir

cuestionario” se envía un POST con el identificador del autorregistro correspondiente. Al recibirse en el servidor este identificador se obtienen las preguntas asociadas a ese autorregistro y se muestran para que el paciente las complete.

Cuando el paciente haya terminado de responder al autorregistro se muestra un botón para proceder al envío de las respuestas a la base de datos, donde se guardarán para su posterior visualización por parte del terapeuta.

En la ilustración 17 se puede ver un ejemplo de la interfaz a la que accede el usuario paciente, con una lista de los autorregistros que se le han asignado, la fecha de realización y la opción de abrir cada uno de ellos para completarlo.

The screenshot shows a web application interface for a patient. At the top, there is a dark navigation bar with a blue 'Menú' button, a 'Perfil' link, and a 'Cerrar sesión' link. Below the navigation bar, the section is titled 'Autorregistros asignados:'. It contains three entries, each in a light gray box. Each entry displays the name of the questionnaire, its scheduled date, and a blue 'Abrir cuestionario' button.

Autorregistro	Fecha programada	Acción
Consumo de cigarrillos	todos los lunes	Abrir cuestionario
Situaciones de estrés	cualquiera	Abrir cuestionario
Control de dolores de cabeza	2020-01-20	Abrir cuestionario

Ilustración 17: Menú aplicación destinada a pacientes.

La siguiente opción disponible en la barra de navegación es la de 'Perfil'. Al igual que en el caso de la web destinada a terapeutas, se mostrarán los datos del paciente (nombre, correo electrónico y contraseña), así como una opción donde cambiar estos datos. También se incluye una opción para eliminar los datos del paciente. En esta opción se eliminará toda la información sobre el paciente, de modo que el terapeuta no podrá enviar autorregistros a este paciente ni volver a visualizar las respuestas que había enviado a los autorregistros que haya completado.

Menú Perfil Cerrar sesión

Nombre: Ejemplo

Cambiar nombre

Correo electrónico: ejemplo@ejemplo.com

Cambiar correo

Si desea cambiar la contraseña, introduzca una nueva contraseña:

Cambiar contraseña

Si desea eliminar su perfil pulse aquí: **(Aviso: se perderan todo los datos y se eliminara toda su información sobre autoinformes completados.)**

Eliminar perfil

Ilustración 18: Menú aplicación destinada a pacientes.

3.5 Encriptación de datos

En cuanto a la encriptación utilizada para garantizar la seguridad de los datos del usuario se optó por encriptar las contraseñas. En principio no debería haber problemas en cuanto a la seguridad debido a que solo el administrador de la base de datos debería poder acceder a las contraseñas almacenadas en la base de datos. Sin embargo, alguien podría obtener las credenciales de acceso del administrador y acceder a la base de datos con las contraseñas y los correos electrónicos asociados. Por ello es necesario realizar la encriptación de las contraseñas.

La primera opción y más simple podría ser utilizar una clave privada que solo tuviera el administrador de la base de datos, sin embargo, nos encontraríamos con el mismo problema que en el caso anterior, debido a que, si alguien obtuviera las credenciales del administrador, también podría obtener su clave privada y con ella descifrar todas las contraseñas de los usuarios de la aplicación web desarrollada en este proyecto.

Por esto, la opción óptima para este proyecto es la encriptación de las contraseñas de los usuarios con funciones hash.

3.5.1 Encriptación con funciones hash

En este tipo de encriptación se usan funciones Hash que reciben como parámetro una cadena de caracteres, en este caso, la contraseña del usuario, y devuelven un código de longitud fija que se corresponde con la contraseña encriptada, pero a partir del cual, no debería poder obtenerse de nuevo la contraseña encriptada.

En PHP existen varios algoritmos para realizar esta encriptación, como pueden ser MD5, SHA1, SHA256 y para utilizarlos disponemos de diversas funciones para ello como por ejemplo “md5”, “sha1”, “hash”. Sin embargo, estos algoritmos han sido ampliamente utilizados desde hace mucho tiempo y hoy en día no se recomienda su uso debido a que ya han sido estudiados y existen las denominadas "rainbow tables" en las que se pueden encontrar numerosas contraseñas y su código hash asociado, por lo que es fácil recuperar una contraseña a partir de su código Hash.

La mejor opción es utilizar la función “password_hash”. Esta función utiliza un algoritmo “blowfish” y devuelve un código hash que se divide en varias partes. En primer lugar, el algoritmo utilizado, en segundo lugar, el coste, después, la denominada "sal" y por último el código hash. El elemento diferencial que hace que esta función sea más segura que las anteriores es la “sal”. Esta “sal” es un código aleatorio que se aplica al resto del código, por lo que dos usuarios que hayan utilizado la misma contraseña no obtendrán el mismo código hash, lo que hace mucho más difícil su desencriptación (Manual password_hash, s.f.).



Ilustración 19: Ejemplo estructura de contraseña encriptada.

Una vez tenemos la contraseña encriptada guardada en la base de datos tenemos que comprobar que el usuario introduce la contraseña correcta cuando quiere acceder a su perfil. Esta comprobación se realiza mediante la función “password_verify”, presente en las librerías de PHP a partir de PHP 5.5. Como hemos visto que “password_hash” devuelve en el código hash la información del algoritmo, el coste y la sal utilizada, la función “password_verify” tiene todos los datos necesarios para comprobar si la contraseña introducida por el usuario al iniciar sesión se corresponde con el hash de la contraseña almacenado en nuestra base de datos.

4 Integración y resultados

4.1 Integración

Una vez se ha desarrollado la aplicación y se han realizado pruebas de uso con éxito en el servidor local utilizando XAMPP se procederá a subir la aplicación a la red para que sea accesible por todos los usuarios que deseen usarla.

Para ello se seguirán las indicaciones disponibles en la página web de XAMPP (Host your Application in the Amazon Cloud with XAMPP and Bitnami, s.f.), que recomienda el uso de Microsoft Azure, Google Cloud Platform y Amazon Web Services (AWS). Entre estas tres opciones la elegida en nuestro caso es la de Amazon Web Services. Las tres opciones permiten pruebas gratuitas, sin embargo, la opción más barata cuando se terminan las pruebas gratuitas a priori es la de Amazon Web Services.

4.1.1 Alojamiento en Amazon Cloud

El primer paso es el de registrarse en AWS. Para ello se completan los datos necesarios, incluyendo una tarjeta de crédito para verificar la identidad del usuario. Se elige el plan básico gratuito y se confirma nuestra cuenta.

El segundo paso es registrarse en Bitnami. Este es un servicio que permite desplegar un servidor en la nube de forma rápida y sencilla. Una vez nos hemos registrado tanto en AWS como en Bitnami, debemos conectar ambas cuentas.

En el panel de AWS accedemos a la opción de “Credenciales de seguridad” y desde ahí pulsamos en “Crear nueva clave de acceso”. Debemos anotar estas claves para introducirlas a continuación en nuestra cuenta de Bitnami. Accedemos a la consola de Bitnami y pulsamos la opción de “Crear servidor”. Al pulsarla se nos requerirán las claves de acceso de Amazon que acabamos de apuntar.

A continuación, seleccionamos la opción de servidores y elegimos la configuración establecida por defecto para provisionar el servidor que alojará nuestra aplicación. Esta configuración nos proporciona un servidor con Ubuntu, 1000 MB de RAM y 10 GB de almacenamiento. Además, debemos seleccionar la aplicación de LAMP en la opción de “añadir aplicación”. Tras confirmar, nuestro servidor estará listo y podremos acceder a él desde cualquier navegador. Para empezar a usar nuestro servidor pulsamos en “conectar”, donde podremos descargar las credenciales SSH necesarias para conectarnos a él.

El siguiente paso es comprobar que tanto PHP como MySQL funcionan correctamente en nuestro servidor. Para poder comunicarnos con nuestro servidor debemos utilizar el programa PuTTY. En este programa configuraremos la conexión utilizando como autorización SSH las credenciales que hemos descargado desde el panel de Bitnami. Una vez comience la conexión aparecerá un terminal donde introduciremos estas instrucciones:

```
cd /opt/bitnami/apache2/htdocs
echo "<?php phpinfo(); ?>" > phpinfo.php
```

De este modo se crea un fichero que nos muestra la configuración de PHP. Si todo ha funcionado correctamente podremos acceder desde nuestro navegador a la dirección `http://[nombre_del_servidor]/phpinfo.php`, donde podremos observar este resultado en el navegador:


<div>PHP Version 7.4.6</div> <div>  </div>	
System	Linux ip-172-31-76-191 4.19.0-9-cloud-amd64 #1 SMP Debian 4.19.118-2 (2020-04-29) x86_64
Build Date	Jun 3 2020 09:19:34
Configure Command	'/bitnami/blacksmith-sandbox/php-7.4.6/configure' '--prefix=/opt/bitnami/php' '--with-imap=/bitnami/blacksmith-sandbox/imap-2007.0.0-f' '--with-imap-ssl' '--with-zlib-dir' '--with-zlib' '--with-libxml-dir=/usr' '--enable-soap' '--disable-rpath' '--enable-inline-optimization' '--with-bz2' '--enable-sockets' '--enable-pcntl' '--enable-exif' '--enable-bcmath' '--with-pdo-mysql=mysqlnd' '--with-mysql=mysqlnd' '--with-png-dir=/usr' '--with-openssl' '--with-libdir=/lib/x86_64-linux-gnu' '--enable-ftp' '--enable-calendar' '--with-gettext' '--with-xmllib' '--enable-fpm' '--with-fpm-user=daemon' '--with-fpm-group=daemon' '--enable-mbstring' '--enable-cgi' '--enable-ctype' '--enable-session' '--enable-mysqlnd' '--enable-intl' '--with-iconv' '--with-pdo_sqlite' '--with-sqlite3' '--with-readline' '--with-gmp' '--with-curl' '--with-pdo-pgsql=shared' '--with-pgsql=shared' '--with-config-file-scan-dir=/opt/bitnami/php/etc/conf.d' '--enable-gd' '--with-pear' '--with-freetype' '--with-jpeg' '--with-zip' '--with-pdo-dblib=shared' '--with-tidy' '--with-ldap=/usr' '--enable-apcu=shared'
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/opt/bitnami/php/lib
Loaded Configuration File	/opt/bitnami/php/etc/php.ini
Scan this dir for additional .ini files	/opt/bitnami/php/etc/conf.d
Additional .ini files parsed	(none)
PHP API	20190902
PHP Extension	20190902
Zend Extension	320190902
Zend Extension Build	API320190902.NTS
PHP Extension Build	API20190902.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk

Ilustración 20: Resultado al acceder a `http://[nombre_del_servidor]/phpinfo.php`.

Para comprobar si MySQL funciona correctamente podemos acceder a la dirección `http://127.0.0.1:8888/phpmyadmin` e ingresar usuario y contraseña definidas en Bitnami. En ella podremos acceder a la misma interfaz a la que se accedía cuando usábamos XAMPP y servidor local para realizar las pruebas de nuestra web. Además, podemos exportar las bases de datos que estuviéramos utilizando en local e importarlas en el servidor que acabamos de desplegar.

El último paso que debemos realizar para completar el proceso es subir el código de nuestra aplicación al servidor. Para ello utilizamos FileZilla. Configuramos la conexión al servidor con el nombre de nuestro servidor, el protocolo SFTP, y el archivo con las claves

privadas que descargamos de Bitnami. En este último paso se mostrarán a un lado los archivos presentes en nuestro ordenador personal y en el otro los archivos del servidor. Arrastrando de uno a otro podremos enviar el código desde nuestro ordenador al servidor y la aplicación estará lista para comenzar a funcionar.

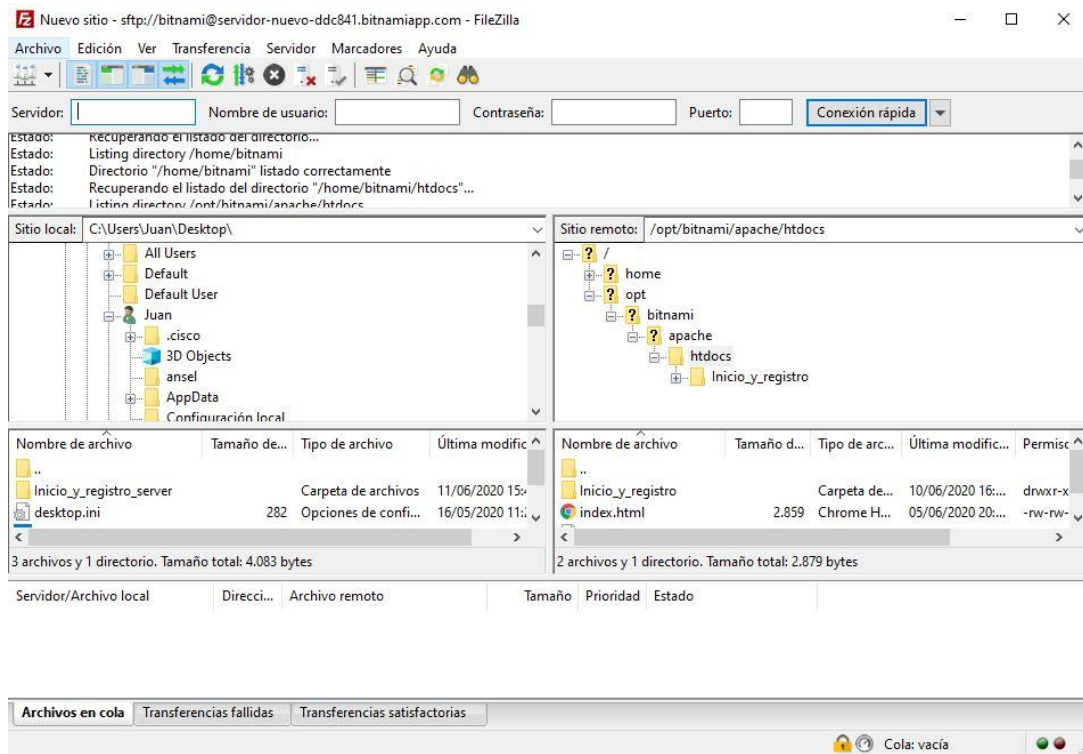


Ilustración 21: Captura Filezilla.

Antes de subir los archivos con el código es importante realizar un cambio en todos ellos. Hasta ahora estábamos realizando las pruebas en el equipo local, por lo que en la conexión al servidor local que permitía realizar las consultas a nuestras bases de datos se introducían el usuario y contraseña que teníamos en phpMyAdmin en ese servidor local. Sin embargo, al subirlo a nuestro servidor de Amazon en la nube, debemos utilizar las credenciales que se nos indican en la consola de Bitnami para nuestro servidor y nuestro nuevo usuario de phpMyAdmin.



Ilustración 22: Credenciales Bitnami.

4.1.2 Página de inicio

Tras haber subido el código a nuestro servidor ya podremos acceder a nuestra página web en el nombre de dominio asignado que podemos observar en el panel de control de nuestro usuario de Bitnami. Para finalizar nuestra página debemos configurar una página de inicio que muestre el objetivo y la utilidad de la página a los usuarios, además de añadir enlaces para redirigirlos a las páginas de registro e inicio de sesión de terapeutas y pacientes.

Autorregistros online

The image displays the 'Autorregistros online' web application. The top section features a green banner for therapists ('Inicia sesión o regístrate como terapeuta') and a blue banner for patients ('Inicia sesión o regístrate como paciente'), each with 'Registro' and 'Inicio de sesion' buttons. Below, the 'Terapeutas' section explains the registration process and shows a sidebar with navigation links. The 'Pacientes' section shows a list of assigned questionnaires for a patient, including 'Consumo de cigarrillos', 'Situaciones de estrés', and 'Control de dolores de cabeza', each with a 'Ver cuestionario' button.

Inicia sesión o regístrate como terapeuta

[Registro](#) [Inicio de sesion](#)

Inicia sesión o regístrate como paciente

[Registro](#) [Inicio de sesion](#)

Terapeutas

Al registrarte como terapeuta podrás acceder a tu perfil, donde podrás crear autorregistros para que sean completados por tus pacientes de forma completamente gratuita.

Podrás agregar a tus pacientes una vez se hayan registrado como pacientes mediante su correo electrónico. Podrás asignarles los autorregistros que desees y ver las respuestas que tus pacientes dan a dichos autorregistros.

Inicio **Perfil**

Autorregistros asignados:

- Autorregistro: Consumo de cigarrillos
Fecha programada: todos los lunes
[Ver cuestionario](#)
- Autorregistro: Situaciones de estrés
Fecha programada: cualquiera
[Ver cuestionario](#)
- Autorregistro: Control de dolores de cabeza
Fecha programada: 2020-01-20
[Ver cuestionario](#)

Pacientes

Al registrarte como paciente, podrás acceder a los autorregistros que te asigne su terapeuta para completarlos.

Para que tu terapeuta pueda agregarte, le debes indicar el correo electrónico con el que te has registrado.

Ilustración 23: Página de inicio.

4.1.3 Cuestiones de interés legal

Debido a que no se van a obtener beneficios económicos con este sitio web ni se van a utilizar anuncios para generar ingresos no es necesario realizar un “Aviso legal”. Sin embargo, se ha realizado un apartado en el que se exponen las condiciones y el email de contacto.

Las condiciones y avisos planteados en este apartado son los siguientes:

- La finalidad del sitio web, es decir, que sirva como herramienta de creación y asignación de autorregistros.
- Que los datos introducidos en la web serán almacenados en bases de datos y solo serán accesibles por el paciente y su terapeuta.
- Que todos estos datos pueden ser borrados en el apartado “Perfil”.
- Que no se puede garantizar la inexistencia de virus u otros elementos que puedan producir inconvenientes en los sistemas del usuario.
- Que la utilización del sitio web es gratuita.
- Que el usuario registrado es mayor de edad.

4.1.4 De HTTP a HTTPS

El último paso para realizar antes de que nuestra web sea realmente usada por cualquiera es garantizar que los datos que se transmiten desde el navegador al servidor lo hacen de forma segura. Para ello es necesario realizar el cifrado a nivel de transporte con SSL. La herramienta utilizada para ello es la herramienta bncert-tool, proporcionada por Bitnami. En la documentación (Learn About The Bitnami HTTPS Configuration Tool, s.f.) de Bitnami podemos encontrar un sencillo tutorial que nos muestra como ejecutarlo para que nuestro sitio web pase de HTTP a HTTPS. Esta herramienta está presente en todas las instalaciones de Bitnami desde el 10 de mayo de 2019. Tras ejecutar los comandos que nos indica la documentación, podremos comprobar que al acceder a nuestro sitio web contamos con certificado válido y nuestra URL comienza por “https”, como se muestra en la Ilustración 24.



Ilustración 24: Certificado SSL.

5 Conclusiones y trabajo futuro

5.1 Conclusiones

En este trabajo de fin de máster se ha implementado una aplicación web de autorregistros destinada a terapeutas psicológicos y sus pacientes. Se ha realizado con una interfaz amigable y cómoda para los usuarios, priorizando la facilidad de uso y la simpleza en su diseño. Para ello se ha comprobado que la forma más fácil de reproducir un entorno de programación web gratuito es utilizar XAMPP para desplegar un servidor local y probar el código desarrollado en él.

Los lenguajes de programación utilizados han sido HTML, CSS, PHP, SQL y JavaScript. HTML proporciona un recurso muy útil y sencillo para compartir información entre el navegador del cliente y el servidor donde está alojada nuestra página web. Estos recursos son los formularios. Por medio de ellos se pueden enviar datos al servidor para que sean procesados mediante PHP. Con PHP también se pueden realizar consultas SQL a nuestras bases de datos para mostrar los resultados a las acciones que realice el usuario de nuestra web.

Una vez desarrollado el código de nuestra web, de entre las numerosas opciones posibles en el mercado para contratar un host y desplegar en él nuestra aplicación, se ha optado por AWS y Bitnami. La mayoría de estos servicios de alojamiento proporcionan versiones de prueba gratuita, pero la razón principal para elegir AWS es que dispone de un periodo gratuito de 12 meses. Por otro lado, el motivo por el que se ha usado Bitnami para desplegar en AWS la aplicación, es que es una de las herramientas más populares para ello y cuenta con una amplia comunidad y documentación, lo que facilita el proceso de buscar la información que necesitemos cuando nos encontremos con un inconveniente.

Tras completar el lanzamiento de la aplicación, podemos concluir que en este proyecto se ha materializado una herramienta muy útil para el seguimiento en patologías psicológicas que actualmente no tiene ninguna opción similar gratuita en el mercado. Con esta herramienta los terapeutas tienen una opción disponible con un funcionamiento muy simple y sencillo para crear autorregistros personalizados y enviárselos a sus pacientes. También podrán visualizar los resultados una vez que los pacientes hayan completado los autorregistros que se les asignen.

Los pacientes mediante esta aplicación web podrán completar los autorregistros que sus terapeutas les asignen de forma cómoda, desde su ordenador, tablet o smartphone (tanto Android como IOS), sin tener que recurrir a métodos anteriores como lápiz y papel teniendo que llevar los autorregistros impresos consigo cuando querían completarlos.

El trabajo realizado en este proyecto se ha dividido por tanto en las siguientes tareas y subtareas:

- 1) **Creación de página web dirigida a terapeutas, con menú intuitivo para creación de autorregistros de evaluación y envío de los mismos a pacientes.**

- a) Instalación de XAMPP. Planificación y creación de la base de datos y estructura de la página web dirigida a los terapeutas.
 - b) Creación de menú de usuario para cada terapeuta en el que introducir las preguntas eligiendo los tipos de posibles respuestas.
 - c) Listado de pacientes, con opción de creación de nuevos casos: asociación de la plantilla de evaluación creada a un paciente o grupo de pacientes, con especificación de la fecha de evaluación.
 - d) Listado de autorregistros creados.
 - e) Listado de resultados de cada paciente con gráficas utilizando la librería de Google Charts disponible de manera gratuita.
 - f) Información sobre el usuario, con posibilidad de cambios en los datos.
- 2) Creación de aplicación web para pacientes, receptora de los autorregistros diseñados por el terapeuta.**
- a) Menú de autorregistros disponibles por el paciente.
 - b) Sistema de envío de respuestas desde el perfil del paciente para su posterior consulta por su terapeuta asignado.
 - c) Información sobre el usuario, con posibilidad de cambios en los datos.
- 3) Despliegue de la aplicación en la nube.**
- a) Contratación de servicios gratuitos de AWS.
 - b) Subida del código necesarios al servidor mediante Bitnami.
- 4) Adopción de medidas de seguridad.**
- a) Encriptación de las contraseñas.
 - b) Configuración de página segura con HTTPS.

5.2 Trabajo futuro

En cuanto al trabajo futuro, el objetivo principal es el mantenimiento de la web, de modo que no quede desactualizada y se atiendan las incidencias de los pacientes y terapeutas que opten por su utilización. Según vaya creciendo el uso, se deben ir escalando los recursos utilizados por la web. Como, por ejemplo, mejorar el servicio de la capa gratuita de AWS y contratar un servidor de mayor capacidad con mejor rendimiento.

Además, se deben tener en cuenta las sugerencias de pacientes y terapeutas para implementar nuevas funcionalidades. A pesar de que el funcionamiento es bastante simple, se deben satisfacer las necesidades de los usuarios si se comprueba que hay confusiones o errores a la hora de utilizarla.

Las mejoras que se plantean para el futuro son las siguientes:

- Visualización de los resultados mediante alguna herramienta que permita al terapeuta realizar dibujos o anotaciones sobre las gráficas y las tablas. Este avance podría venir de la mano de la implementación de una aplicación móvil en Android e IOS. De tal forma que permita a los pacientes realizar los autorregistros desde ella en vez de desde una página web. Así, el terapeuta podría usar su tablet para realizar anotaciones durante la consulta y señalar al paciente los aspectos más importantes sobre las gráficas de resultados. El desarrollo de una aplicación móvil también permitiría la activación de notificaciones como recordatorio para los pacientes de que tienen que completar un autorregistro. También se podría notificar al terapeuta cada vez que un usuario complete un autorregistro.
- Otra alternativa para la mejora en la visualización de los resultados sería permitir la descarga de las tablas y gráficas generadas en el apartado de resultados en formato jpg, o la descarga de los propios resultados en formato CSV.
- Las posibilidades de mejora están abiertas a las sugerencias del equipo del profesor Álvaro Sánchez López de la UCM. El protocolo de pruebas a realizar por este equipo implicaría varios usuarios en el rol de terapeutas y varios en el rol de pacientes, con testeo de cada uno de los tipos de preguntas. En cuanto a diseño y programación de envíos por parte de los terapeutas, se probaría el acceso a los autorregistros en las fechas programadas por los pacientes, control de acceso a datos de modo correcto por parte de los terapeutas, y funcionalidad de la última parte de visualización gráfica de resultados.
- Una vez se tenga listo el programa final, se implementaría con terapeutas reales y sus pacientes, a partir de septiembre, y se haría una evaluación de usabilidad por cada uno de los perfiles, mediante entrevistas cualitativas con cada grupo y varias escalas de usabilidad que se suelen aplicar en la validación de aplicaciones y páginas web para los pacientes:
 - System Usability Scale (SUS)
 - User Engagement Scale (UES)
- Se podrían redactar una serie autorregistros plantilla sobre los problemas más habituales de los pacientes que fueran accesibles por todos los terapeutas.
- La aplicación web podría evolucionar hasta una aplicación no solo limitada a la realización de autorregistros. En ella se podrían gestionar también las citas con los pacientes, enviando recordatorios de cita a pacientes y terapeutas, pudiendo modificarlas o eliminarlas.
- Adición de opciones para que el terapeuta pueda diseñar, no solo autorregistros, sino ejercicios conductuales para el paciente.

- Cambiar el dominio de nuestra página web para que sea más fácilmente accesible por los usuarios. También, mejorar el posicionamiento SEO con el uso de palabras clave para que aparezca en los primeros lugares al realizar búsquedas.
- Se podrían implementar traducciones para que la web fuera accesible por más usuarios.

6 Bibliografía

- Bringing MySQL to the web.* (s.f.). Obtenido de <https://www.phpmyadmin.net/>
- Documentación mysqli_connect.* (s.f.). Obtenido de <https://www.php.net/manual/es/mysqli.construct.php>
- Documentación mysqli_query.* (s.f.). Obtenido de <https://www.php.net/manual/es/mysqli.query.php>
- Documentación mysqli_real_escape_string.* (s.f.). Obtenido de <https://www.php.net/manual/es/mysqli.real-escape-string.php>
- Documentación uso básico \$_SESSION.* (s.f.). Obtenido de <https://www.php.net/manual/es/session.examples.basic.php>
- Eguíluz Pérez, J. (2009). *Introducción a JavaScript*.
- Fernández-Ballesteros, R. (2011). *Evaluación Psicológica: Conceptos, métodos y estudio de casos*. Madrid: Piramide.
- Ferrer, J., García, V., & García, R. (s.f.). *Curso completo de HTML*.
- Form submission. HTML 4.01 Specification.* W3C. (1999). Obtenido de <https://www.w3.org/TR/html401/interact/forms.html#submit-format>
- Host your Application in the Amazon Cloud with XAMPP and Bitnami.* (s.f.). Obtenido de <https://www.apachefriends.org/docs/hosting-xampp-on-aws.html>
- Jason Gilmore, W. (2010). *Beginning PHP and MySQL: From Novice to Professional*. Apress.
- Learn About The Bitnami HTTPS Configuration Tool.* (s.f.). Obtenido de <https://docs.bitnami.com/aws/how-to/understand-bncert/>
- Manual password_hash.* (s.f.). Obtenido de <https://www.php.net/manual/es/function.password-hash.php>
- Naglieri, J., Drasgow, F., Schmit, M., Handler, L., Prifitera, A., Margolis, A., & Velasquez, R. (2004). Psychological Testing on the Internet. *American Psychologist*.
- Overview of the MySQL Database Management System.* (s.f.). Obtenido de <https://dev.mysql.com/doc/refman/8.0/en/what-is.html>
- Pérez Álvarez, M., & Fernández-Ballesteros, R. (1994). *Evaluación conductual hoy*. Madrid: Piramide.
- Schulz, R. (2008). *Diseño Web con CSS*. Marcombo.

Glosario

XAMPP	Distribución de Apache completamente gratuita y fácil de instalar que contiene MariaDB, PHP y Perl.
Bitnami	Biblioteca de instaladores o paquetes de software para aplicaciones web.
AWS	Colección de servicios de computación en la nube ofrecidos a través de Amazon
Autorregistros	Cuestionarios de distintos tipos que intentan recopilar información sobre las conductas que tienen relación con problemas psicológicos.
SSH	Protocolo cuya función es el acceso remoto a un servidor por medio de un canal seguro.
SSL	Protocolo de seguridad de la capa de transporte.